Mining of Sequential Patterns applied to the Prediction of Protein Folding

J. Quintana-Zaez, MSc.[1], Héctor R. Velarde-Bedregal, Dr.[2],
Guillermo Calderón-Ruiz, Dr.[2], and Cosme E. Santiesteban-Toca, Dr.[3]
[1]Universidad de Ciego de Ávila, Cuba. Facultad de Informática, julioq@informatica.unica.cu
[2]Universidad Catolica de Santa Maria, Perú. Facultad de Ciencias e Ingenierías Físicas y Formales,
hvelardeb@ucsm.edu.pe, gcalderonr@ucsm.edu.pe
[3]Instituto Tecnológico de Ciudad Cuauhtémoc, México. Departamento de Postgrado, csantiestebantoca@gmail.com

Abstract*– Sequence mining consists of finding statistically relevant patterns in data collections represented sequentially. These, are an important type of data, where it matters the order that occupy the elements in the set and that finds a wide range of applications in Bioinformatics and Computational Biology. The prediction of protein structures is one of these applications. Where, a protein is no more than a sequence of amino acids forming patterns known as alpha helices, beta sheets and turns. For purposes of our investigation, these collections or secondary structures would be the itemsets, while the amino acids that make up the entire sequence, the items. Despite multiple attempts to predict protein folding, the algorithms developed to date only reach a 35% effectiveness. That is why we propose SPMCcm, an algorithm based on the prediction of frequent sequences and a scheme of classifiers. Which uses the information provided by the amino acid sequence, in two stages. Where, the first stage learns of the interactions between the secondary structures of the proteins, which it extracts as frequent sequences or itemsets. Meanwhile, the second stage learns of the interaction between the amino acids present in the interacting structures or items. The experimental evaluation showed that SPMCcm behaves in a similar way, independently of the base classifier used, reaching accuracies in the prediction of up to 48%, higher than the 35% reported by the literature, without using large computational resources and possessing explanatory capacity.

***Keywords*— Mining sequential patterns, Protein folding, Contact maps, Classification schemes.**

Digital Object Identifier (DOI):
http://dx.doi.org/10.18687/LACCEI2019.1.1.37
ISBN: 978-0-9993443-6-1 ISSN: 2414-6390

Minería de Patrones Secuenciales aplicada a la Predicción del Plegamiento de Proteínas

J. Quintana-Zaez, MSc.[1], Héctor R. Velarde-Bedregal, Dr.[2],
Guillermo Calderón-Ruiz, Dr.[2], and Cosme E. Santiesteban-Toca, Dr.[3]
[1]Universidad de Ciego de Ávila, Cuba. Facultad de Informática, julioq@informatica.unica.cu
[2]Universidad Catolica de Santa Maria, Perú. Facultad de Ciencias e Ingenierías Físicas y Formales,
hvelardeb@ucsm.edu.pe, gcalderonr@ucsm.edu.pe
[3]Instituto Tecnológico de Ciudad Cuauhtémoc, México. Departamento de Postgrado,
csantiestebantoca@gmail.com

Resumen– ***La minería de secuencias consiste en encontrar patrones estadísticamente relevantes en colecciones de datos representados de forma secuencial. Éstos, son un importante tipo de datos, donde importa el orden que ocupan los elementos en el conjunto y que encuentra una amplia gama de aplicaciones en la Bioinformática y la Biología Computacional. La predicción de estructuras de proteínas es una de estas aplicaciones. Donde, una proteína no es más que una secuencia de aminoácidos formando patrones conocidos como hélices alfa, láminas beta y giros. Para efectos de nuestra investigación, estas colecciones o estructuras secundarias serían los itemsets, mientras que los aminoácidos que conforman la totalidad de la secuencia, los ítems. A pesar de múltiples intentos por predecir plegamiento de las proteínas, los algoritmos desarrollados a la actualidad solo alcanzan un 35% de efectividad. Es por ello que proponemos SPMCcm, un algoritmo basado en la predicción de secuencias frecuentes y un esquema de clasificadores. El cual emplea la información brindada por la secuencia de aminoácidos, en dos etapas. Dónde, la primera etapa aprende de las interacciones entre las estructuras secundarias de las proteínas, las cuales extrae como secuencias frecuentes o itemsets. Mientras, que la segunda etapa aprende de la interacción entre los aminoácidos presentes en las estructuras interactuantes o ítems. La evaluación experimental demostró que SPMCcm se comporta de forma similar, con independencia del clasificador base empleado, alcanzando precisiones en la predicción de hasta un 48%, superiores al 35% reportado por la literatura, sin emplear grandes recursos computacionales y posee capacidad explicativa.***

Palabras Clave: minería de patrones secuenciales, plegamiento de proteínas, mapas de contacto, esquemas de clasificación.

I. Introducción

Uno de los mayores retos de la actualidad es desarrollar un mundo sostenible y sustentable, logrando un equilibrio positivo en lo social, económico y ambiental. Una manera para poder conseguirlo es tener en cuenta el estudio de las investigaciones sobre el cambio climático y el descubrimiento de nuevos y diferentes tipos de fuentes de energía, así como controlar y obtener el mayor aprovechamiento de las actuales.

La tarea de clasificación de secuencias consiste en inducir una función de selección, que genere clasificaciones para secuencias de datos a partir de secuencias de entrenamiento. Usualmente esto se obtiene mediante la composición de alguna técnica general de clasificación, con los criterios de selección adecuados. La idea es que secuencias de datos con clasificaciones similares estén estrechamente relacionadas.

La mayoría de los algoritmos implementados para el minado de secuencias frecuentes, utilizan tres tipos diferentes de enfoques de acuerdo a la forma de realizar el conteo de frecuencia a los patrones secuenciales candidatos:

1. Basados en la propiedad A priori [1]. Esta propiedad fue introducida en el minado de reglas de asociación y se basa en que si un patrón es frecuente entonces cualquier subpatrón de él también será frecuente. Esto permite reducir el espacio de búsqueda en el proceso de generación de candidatos. Basado en esta estrategia se presentaron algoritmos como el AprioriAll y el AprioriSome [2] y el algoritmo GSP (Patrón secuencial generalizado) [3].
2. Basados en la reducción del tamaño del conjunto de datos explorados, sustituyendo la fase de generación de candidatos por la realización de proyecciones y técnicas de crecimiento de patrones sobre los datos iniciales. Estos algoritmos siguen una filosofía de “divide y vencerás”, actualizando recursivamente los conjuntos de secuencias frecuentes encontrados. A este grupo pertenecen algoritmos como el FreeSpan [4] y el PrefixSpan.
3. Basados en algoritmos que almacenan solamente las estructuras conocidas como listas de identificadores o de ocurrencias, que describen la ubicación de cada secuencia en la colección de datos. En este grupo se aplican técnicas como la transformación de los datos a formato vertical.

Esta técnica tiene una gran aplicación en muchos escenarios como pronósticos de cambios climáticos y reconocimiento de patrones; donde se necesitan realizar predicciones de comportamientos, basándose en registros de datos históricos. Por otra parte, también encuentra una amplia gama de aplicaciones en las ciencias de la vida [5].

Ejemplos típicos de secuencias frecuentes, dentro de la Bioinformática, son las secuencias genéticas como las de ADN y ARN. Un caso menos trivial es la predicción de la estructura tridimensional de las proteínas. El cual se ha convertido en uno de los problemas más interesantes dentro de la Bioinformática y la Biología Computacional [6]. Debido a que es la base para el diseño de nuevos fármacos, la obtención de productos naturales, entre otros. Además, algunas de las enfermedades más agresivas que afectan la salud humana como el Alzhéimer

Digital Object Identifier (DOI):
http://dx.doi.org/10.18687/LACCEI2019.1.1.37
ISBN: 978-0-9993443-6-1 ISSN: 2414-6390

están estrechamente relacionadas a problemas en el plegamiento de las proteínas, las cuales no son más que secuencias ordenadas de aminoácidos [7].

A pesar de múltiples intentos por elucidar el proceso de plegamiento de las proteínas, los algoritmos desarrollados a la actualidad solo logran predecir las estructuras tridimensionales con una efectividad del 35% para todas las proteínas. Lo cual demuestra claramente que los predictores basados en aprendizaje automático mejoran la eficiencia de los predictores estadísticos, pero todavía no presentan una precisión suficiente para la predicción de la estructura de las proteínas. Por otra parte, la mayoría de los algoritmos de predicción de estructuras de proteínas presentan un alto grado de complejidad computacional y emplean técnicas cuyas bases de conocimiento no son interpretables, convirtiéndolos en cajas negras.

El empleo de la minería de secuencias, parece ser una solución viable a los problemas anteriores. Sin embargo, es necesario considerar que los algoritmos de predicción de secuencias frecuentes suelen recorrer la base de datos varias veces y generan un gran conjunto de candidatos para patrones pequeños. Lo cual significa que para una proteína pequeña (péptido), de 100 amino ácidos, es necesario considerar alrededor de 10^{30} secuencias.

Es por ello que, en la presente investigación nos plantemos como objetivo diseñar un algoritmo que emplee técnicas de predicción de secuencias frecuentes en conjunción con esquemas de clasificadores, a partir de la información brindada por la secuencia de aminoácidos, que permita predecir el plegamiento de las proteínas con una efectividad similar o superior a los algoritmos del estado del arte, sin emplear grandes recursos computacionales y que posea capacidad explicativa.

II. Trabajos Relacionados

En las últimas décadas, se han desarrollado diversos métodos para la predicción de contactos entre residuos, entre los cuales se encuentran métodos basados en redes neuronales [8], [9], máquinas de soporte de vectores [10], [11], modelos ocultos de Markov [12], Árboles de decisión [13], aprendizaje profundo (o Deep learning) [14], algoritmos genéticos [15], [16] y más recientemente, la combinación de clasificadores [autoRef].

Un multiclasificador es un conjunto de clasificadores que combinan sus predicciones siguiendo un determinado esquema, con el fin de obtener una predicción más fiable que la que normalmente serían capaces de obtener en solitario. La combinación de clasificadores ha sido abordada en la literatura a través de distintos términos, entre ellos: ensamblados (ensembles) [17], [18]; modelos múltiples (multiple models) [19], [20]; sistemas de múltiples clasificadores (multiple classifier systems) [21]; combinación de clasificadores (combining classifiers) [22]; integración de clasificadores (integration of classifiers) [23]; mezcla de expertos (mixture of experts) [24], [25]; comité de decisión (decision committee) [26]; comité de expertos (committee of experts); fusión de clasificadores (classifier fusion) [27], [28] y aprendizaje multimodelo (multimodel learning).

Existen varios métodos propuestos para la construcción de multiclasificadores. Éstos, podrían agruparse en aquellos que utilizan un único algoritmo de construcción de clasificadores base y en aquellos que combinan varios algoritmos de construcción de clasificadores para crear un único clasificador final.

En el primer grupo se encuentran: Bagging [29], Random Forests [30], [31], Random Subspaces [32] y Boosting [33]. Dentro de los algoritmos adaptativos de la familia Boosting, dos de los más empleados son el AdaBoost (AdaBoostM1 y AdaBoostM2) [34] y el MultiBoosting [35].Estos métodos logran la diversidad variando los datos de entrenamiento que procesan, ya sean instancias o atributos. En el segundo grupo se encuentran: Cascading [36], Stacking [37] y Grading [38]. Estos métodos, los datos que se procesan incluyen la información de salida del nivel anterior.

Además de estos algoritmos de multiclasificación clásicos, existen otros muchos esquemas de ensamblado de clasificadores, entre los que se encuentran: Modelo de selección de clases (MCS) [39], divide el conjunto de clases en subespacios a través de reglas obtenidas empíricamente. Asigna a cada espacio un clasificador de tres posibles (árboles de decisión, función discriminante o clasificador basado en instancias). Mezcla de expertos (ME) [40], igual que MCS pero los subespacios pueden estar solapados entre sí. Otro clasificador combina las salidas de los expertos. Existe la variante jerárquica (HME) en que los espacios se descomponen recursivamente en nuevos subespacios. Árbitros de árboles (AT) [41], parte de un número de particiones disjuntas del conjunto de entrenamiento. Con cada una se entrena un clasificador base del primer nivel. Estos clasificadores se emparejan y por cada pareja se entrena otro de nivel superior que actúa como árbitro, que también es un árbol. Este esquema se extiende recursivamente. Combinación de árboles (CA) [41], similar a AT, solo que los clasificadores que no son hojas se entrenan en las salidas del nivel anterior. Emplean las mismas reglas que cascading o satcking. NBTree [42], multiclasificador es un árbol que en sus hojas hay un clasificador Naïve Bayes.

Recientemente, se han desarrollado técnicas que emplean una representación espacial de la vecindad entre estructuras secundarias, residuos e incluso fragmentos de la proteína. Las cuales predicen los las interacciones entre las estructuras secundarias, conocidos como “coarse contact maps”, en lugar de los contactos entre residuos [43], [44], [45], [46].

La predicción de patrones secuenciales en la estructura de una proteína, es un problema altamente desbalanceado. Con múltiples retos como son: muestras, a menudo, poco representativas de estos patrones, alto nivel de solapamiento (patrones o interacciones entre los mismos, que codifican para estructuras diferentes), outliers y alta dimensionalidad (lo cual ocasiona la

generación de un gran número de vectores) [47]. Por otra parte, los clasificadores tienden a detectar mejor los patrones más frecuentes o mayoritarios y a penalizar los patrones minoritarios [48]. Razón por la cual la mayoría de los algoritmos conocidos para la predicción de estructuras, así como para la extracción de patrones secuenciales, tienden a fallar.

III. Algoritmo Propuesto

A. *Fundamentos Bilógicos*

La organización de una proteína se encuentra definida por cuatro niveles estructurales comúnmente denominados como estructuras: primaria, secundaria, terciaria y cuaternaria [49]. Sin embargo, una proteína no es más que una secuencia de aminoácidos (Fig. 1a). Donde, suelen encontrarse colecciones de aminoácidos formando patrones conocidos como hélices alfa, láminas beta y giros. Para efectos de nuestra investigación, estas colecciones o estructuras secundarias serían los ***itemsets***, mientras que los aminoácidos que conforman la totalidad de la secuencia, los ***ítems***.

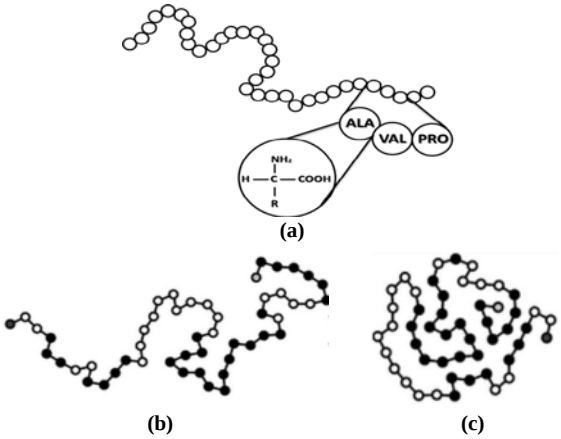


Fig. 1 Abstracción de una proteína, donde: (a) representa la secuencia de aminoácidos que la conforma, también denominada como primera dimensión; (b) representa las regiones de secuencias frecuentes reconocidas como estructuras secundarias o segunda dimensión; (c) representa la proteína plegada o tercera dimensión y la responsabilidad de estas secuencias en el plegamiento de la proteína.

Es común que algunas zonas de la proteína tengan entidad estructural independiente, y a menudo funciones bioquímicas específicas, como, por ejemplo, alguna actividad catalítica. Estas zonas son conocidas como: dominios, motivos o elementos conformacionales. Muchos dominios son únicos y proceden de una secuencia única de un gen o una familia génica, pero en cambio otros aparecen en una variedad de proteínas. La estructura de los motivos a menudo pueden ser relativamente simples, consistiendo en solo unos pocos elementos, por ejemplo, las hélice-giro-hélice, alfa-alfa (dos hélices alfa unidas por un giro), beta-beta (dos láminas beta unidas por un giro), beta-alfa-beta (lámina beta unida por un giro a una hélice alfa que esta a su vez unida a otra lámina beta por otro lazo) o estructuras más complejas, como el motivo llamado de "Llave Griega" ("Greek key"), o el Barril Beta ("beta-barrel").

Las relaciones entre los aminoácidos a través de la cadena de la proteína (Fig. 2a), juegan un importante papel en el proceso de plegamiento y estabilización de la misma de la misma [50]. Con el objetivo de conocer cómo interactúan estas estructuras frecuentes, el algoritmo realizará un minado en sus representaciones como mapas de contacto (Fig. 2b). Donde se analizarán las interacciones entre estas estructuras y su responsabilidad a corto, medio y largo alcance, dentro del proceso de plegamiento de la proteína.

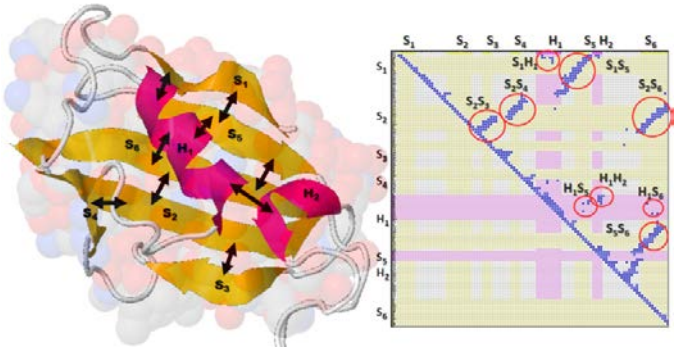


Fig. 2 Interacción entre los patrones secuenciales (**itemsets**) de la proteína 1A9H, hélices alfa y láminas beta, (a) y su representación en una matriz bidimensional denominada mapa de contactos (b)

Un mapa de contacto no es más que una representación en una matriz cuadrada de las interacciones entre los aminoácidos presentes en la secuencia y los valores en las celdas representan la interacción entre éstos (Fig. 2b). Esta información contribuye a entender cómo se organizan los residuos en el espacio y a descifrar los procedimientos de pliegue de las proteínas [51], [52]. De igual manera, la información del mapa de contacto se emplea en la predicción de estructuras desconocidas y funciones de las proteínas [53].

B. *Arquitectura del Algoritmo propuesto*

El predictor propuesto, parte de la base de que las estructuras secundarias y sus vecindades son las responsables del 90% de los contactos entre residuos. Por tanto, si se predicen dichas interacciones, se puede elevar el nivel de precisión de la predicción de los mapas de contactos de proteínas. Por tal motivo fue denominado ***SPMCcm*** (de sus siglas en inglés "***Sequential Patterns based Multiple Classifier for Contact Map prediction***")

Múltiples estudios demuestran que los esquemas de clasificación pueden resolver o reducir el efecto del desbalance, mencionado con anterioridad. Tomando en cuenta la naturaleza desbalanceada del problema a resolver, es que proponemos el empleo de un esquema de clasificación basado en dos etapas. Dónde, la primera etapa aprende de las interacciones entre los ***itemsets*** (secuencias frecuentes o estructuras secundarias). Mientras, que la segunda etapa aprende de la interacción entre los

ítems involucrados (aminoácidos presentes en las estructuras interactuantes).

La característica más deseable de un esquema de clasificación es la diversidad. La cual es la clave para incrementar el desempeño de éstos. Es por ello que, en el esquema propuesto (Fig. 3), cada etapa de clasificación se compone de varios clasificadores base los cuales son entrenados empleando diferentes conjuntos de datos y empleando una técnica de muestro con reposición.

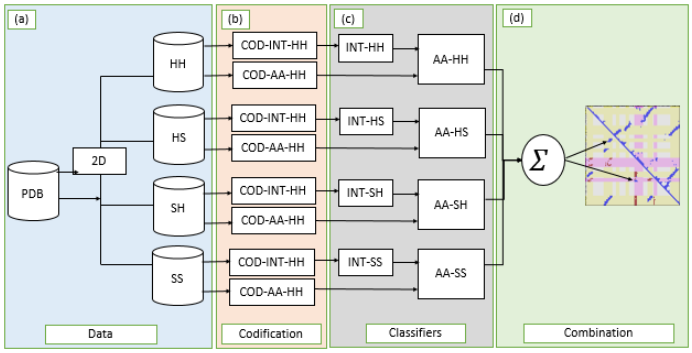


Fig. 3 Esquema de clasificación propuesto: (a) nivel de datos; (b) nivel de codificación; (c) nivel de clasificadores base; (d) nivel de combinación de los resultados.

La Fig. 3, muestra una representación esquemática del esquema de combinación propuesto, el cual se encuentra separado en cuatro niveles. En el primer nivel (datos), se crea un conjunto de datos que contiene las interacciones entre las secuencias frecuentes. Este nivel entrega un subconjunto de datos para cada tipo de interacción entre secuencias frecuentes, tomando en consideración el orden de las mismas, por lo que se conforman de la siguiente forma: hélices-hélices, hélices-láminas, láminas-hélices, láminas-láminas. En el segundo nivel (codificación), se crean los vectores de características que describen las interacciones ente los ***itemsets*** entregados por el nivel de datos. Posteriormente, en el tercer nivel (clasificación), se realiza el entrenamiento de los clasificadores base. Los cuales son los encargados de aprender de las interacciones ente los ***itemsets*** (primera etapa) y de las interacciones entre los ***ítems*** incluidos en estos ***itemsets*** (segunda etapa).

Finalmente, en el cuarto nivel (combinación de resultados), se llega a un consenso entre las predicciones de las etapas del nivel anterior, para construir el mapa de contacto de la proteína.

En este nivel, pueden ser empleados diversos métodos para la combinación de los resultados. Alguno de los más comunes son el voto mayoritario, el voto mayoritario ponderado, Naive Bayes, entre otros. Los cuales pueden clasificarse como métodos de nivel de rankeo o de nivel de medición. Para los efectos del esquema propuesto, se optó por una combinación a nivel de medición, debido a que se conoce previamente la efectividad de cada clasificador base en cada etapa (Ecuación 1).

$$E = \sum_{j=1}^{P} w_j * e_j \qquad (1)$$

Dónde, ***E*** es la estimación final, w_j es el peso asignado por el clasificador de la primera etapa, e_j, la estimación del clasificador de la segunda etapa y ***P***, el tipo de interacción. Si se toma en cuenta que el clasificador de la primera etapa retorna la probabilidad de interacción entre los dos tipos de ***itemsets*** para los cuales está especializado, mientras el resultado del resto de los clasificadores es multiplicado por cero, entonces este mecanismo de combinación sería equivalente a un método de selección del clasificador apropiado para cada tipo de ***itemsets***.

Adicionalmente, en este nivel, se genera un conjunto de reglas que son capaces de describir los patrones encontrados, como resultado de las interacciones entre las secuencias frecuentes presentes en la proteína.

IV. Análisis y Discusión de los Resultados

A. Selección de los Datos

Con el objetivo de evaluar el desempeño del modelo propuesto y de compararlo con otros algoritmos, fueron seleccionadas bases de datos propuestas en el estado del arte. Para ello, se tomaron en cuenta 90 proteínas seleccionadas de la base de datos de proteínas (PDB), mediante el empleo de a herramienta PDBSelect. Para esta selección, se tomaron en cuenta proteínas con más de cuatro aminoácidos, sin interrupciones en su cadena peptídica, y con longitudes no superiores a 400 (número ***items***).

TABLA 1.
DESCRIPCIÓN DE LOS ATRIBUTOS EMPLEADOS EN EL VECTOR DE RASGOS. CADA ESTRUCTURA CONSIDERADA CUENTA CON UN CONJUNTO DE ESTOS ATRIBUTOS

Nombre del rasgo	Descripción	Tipo	Entr.
Hidrofobicidad	Promedio de la distribución del perfil de hidrofobicidad descrito por Kyte-Doolittle [30]	Numérico (# Hidrofóbicos, # Hidrofílicos)	2
Polaridad	Promedio de la distribución del perfil de polaridad basado en la escala de Klein [31]	Numérico (# Polares, # No-Polares, # Ácidos, # Básicos)	4
Átomos	Promedio de la distribución del perfil de carga basado en la escala de Klein [31]	Numérico (# Hidrogeno, # Nitrógeno, # Oxigeno, # Carbón y # Sulfuro)	5
Frecuencia de residuos	Promedio de la distribución del perfil de residuos de la estructura.	Numérico (Frecuencia AA)	20
Masa	Promedio de la distribución del tamaño de los aminoácidos	Numérico (# Grandes, # Pequeños)	2
Tamaño	Cantidad de residuos dentro de la estructura secundaria	Numérico	1
Separación	Numero de estructuras entre las estructuras objetivo	Numérico	1
Clase	Clase (Contacto o No-Contacto)	Nominal	1

En la Tabla 1 se muestra una base de datos con diferentes distribuciones de ***itemsets*** (clases estructurales de proteínas): hélices (Alpha), hélices+láminas (Alpha+Beta), hélices/láminas

(Alpha/Beta), láminas (Beta) [53]. Adicionalmente, fue agregada una base de datos más, "Hepatitis C" [54], con el objetivo de realizar la evaluación externa del algoritmo en un caso real.

Para la conformación de los vectores que describen cada uno de los ***itemsets***, fueron empleadas diferentes características de cada uno de los ***ítems*** que lo conforman y que tienen responsabilidad en el proceso de plegamiento de las proteínas. Entre las que destacan: el grado de hidrofobicidad, la polaridad, carga, entre otros (Tabla 1). Para cada una de las etapas se construyó un vector diferente, los cuales describen las interacciones entre ***itemsets*** y las interacciones entre ***ítems*** de ***itemsets*** interactuantes.

B. Resultados Experimentales

Para la evaluación experimental, primeramente, se realizó un análisis de los clasificadores base empleados y se seleccionó la mejor estrategia. Seguido, se analizó es desempeño del algoritmo en diferentes conjuntos de proteínas. Posteriormente se aplicó el método propuesto a un dominio real de aplicación. Finalmente se hace un análisis del multi-clasificador propuesto con respecto a los clasificadores del estado-del-arte. Para evaluar los resultados del esquema experimental fueron empleadas las métricas, ***precisión*** y ***sensibilidad***, las cuales representan los contactos predichos que son positivos y la proporción de contactos positivos que son predichos, respectivamente. También se emplea ***Fm*** (*F-measure*) que es una media aritmética entre la precisión y la sensibilidad [19].

Selección de clasificadores base

En este estudio solo se consideraron árboles de decisión [53], dada las características de los datos, los que presentan un alto nivel de desbalance que puede alcanzar niveles de desbalance entre clases de 9/1 (No-Contacto/Contacto). Además, se desea obtener modelos que permitan explicar que sucede en el proceso de plegamiento de las proteínas. Los algoritmos empleados son implementaciones de Weka para J48 (-C 0.25 -M 2), RandomForest (RF, -I 150 -K 30), RandomTree (RT, -K 0 -M 1.0), REPTree (-M 2 -V 0.001 -N 3 -S 1 -L -1 -P), ADTree (-B 15 -E -3), BFTree (-S 1 -M 2 -N 5 -C 1.0 -P POSTPRUNED), LMT (-I -1 -M 10 -W 0.0), FT (-I 15 -F 0 -M 15 -W 0.0), NBTree [55].

Para evaluar el resultado de los algoritmos empleados en la selección de clasificadores base se empleó el siguiente procedimiento. Donde primeramente se seleccionaron aleatoriamente 10 proteínas (pdb1c9h-107, pdb1e8i-117, pdb1euv-221, pdb1g24-211, pdb1io1-395, pdb1eoe-100, pdb1qi7-253, pdb1g62-224, pdb1ezv-430, pdb1hqm-223, pdb1fwk-296) para conformar una base de datos de prueba. Seguidamente para cada uno de los clasificadores se aplicó un proceso de selección de parámetros óptimos para el problema. Donde, los algoritmos se entrenaron y probaron con todo el conjunto de proteínas, mediante un proceso de validación cruzada (*Cross-validation*, k=10). Finalmente, para cada algoritmo se seleccionó la mejor combinación de parámetros, los mejores resultados obtenidos se muestran en la Tabla 2.

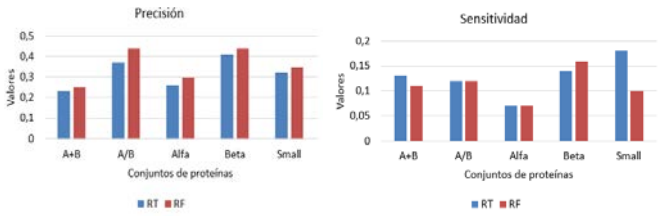
TABLA 2.
RESULTADOS EXPERIMENTALES DE LA SELECCIÓN DE CLASIFICADORES BASE Y SUS PARÁMETROS

Métricas	**LMT**	**FT**	**RF**	J48	**ADT**	**BFT**	**RT**	**REPT**	**NBT**
Precisión	0,48	0,29	0,43	0,48	*0,51*	***0,52***	0,30	*0,51*	0,47
Sensitividad	*0,42*	0,33	***0,71***	0,38	*0,45*	*0,41*	0,29	*0,46*	*0,50*
Fm	0,44	0,30	***0,51***	0,42	*0,47*	*0,46*	0,28	*0,48*	*0,47*

La Tabla 2, muestra los resultados de la selección de clasificadores base, donde se puede apreciar que los valores obtenidos para la presión por los métodos empleados esta entre 0,26 y 0,52 con un promedio de 0,45%. En cuanto a sensibilidad los valores están entre 0,29% y 0,71% con promedio de 0,44%. Para la medida Fm los valores observados están entre 0,27% y 0,51% con promedio 0,43%. Analizando la sensibilidad todos los valores superan el 30%. Además, **RF** sobrepasa al resto de los métodos empleados en el estudio en un 30% más, como promedio, y exhibe la mejor relación entre precisión y sensibilidad con un Fm del 51%. Por otra parte, **RT** es el clasificador con el peor desempeño, alcanzando apenas un 28% de Fm y los valores más pobres de precisión y sensibilidad. Tomando en consideración que se desea evaluar el desempeño del esquema de clasificación propuesto, fueron seleccionados para esta experimentación **FR** y **RT**, el mejor y el peor clasificador base respectivamente.

Dominio de aplicación

Para analizar el desempeño del método propuesto en diferentes dominios de aplicación se empleó un conjunto de 47 proteínas, dividido por clases estructurales (Alfa, Beta, A+B y A/B).



(a) **(b)**

Fig. 4. Gráfico que muestra la precisión (a) y la sensibilidad (b) de los algoritmos en proteínas con diferentes clases estructurales.

La Fig. 4, muestra el desempeño de los mejores clasificadores base frente a estructuras agrupadas según sus clases estructurales. Como se puede apreciar el método de mejor desempeño en cuanto a precisión es ***SPMCcm_RF***. Aunque el desempeño es superior a ***SPMCcm_RT***, la diferencia solo es marcada en el conjunto de proteínas A/B. En el resto de los conjuntos solo es ligeramente superior. El valor más elevado obtenidos es 0,44 y se repite para los conjuntos A/B y Beta. Resultado que se mantiene de acuerdo a los estudios revisados en el estado del arte, donde para estos conjuntos de proteínas, el promedio de los resultados con respecto a Alfa o A+B la mayoría de las veces superior.

Cuando analizamos la sensibilidad podemos observar que los valores más bajos se obtienen para el conjunto de proteínas Alfa. Dato que concuerda con estudios revisados anteriormente. Para el conjunto de proteínas Beta el recuerdo es cerca del 0,15. En cambio, para las proteínas pequeñas el método ***SPMCcm_RT*** supera claramente al ***SPMCcm_RF*** aunque en precisión los valores son similares. Debido a que en cuanto a la sensibilidad general no se puede llegar a una conclusión clara observando los datos, se decidió aplicar una prueba no paramétrica de *Wilcoxon Signed-Rank* con α=0.05 tomando como muestras los valores para la sensibilidad. Como resultado, se obtuvo un p-value=0.8875, lo que sugiere que no existen diferencias significativas entre los resultados de los algoritmos.

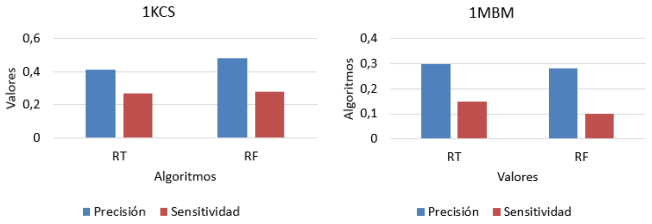
Validación externa (aplicación real)

Para realizar la validación externa de esquema de clasificación propuesto, se analizó su comportamiento en una aplicación real con el conjunto de proteínas del virus de Hepatitis C. Donde se utilizó una parte del conjunto de proteínas para entrenar y la otra parte para probar los clasificadores. Las proteínas empleadas para probar fueron 1KCS y 1MBM, el resto del conjunto se destinó al entrenamiento (Tabla 3).

TABLA 3.
RESULTADOS DE LA PREDICCIÓN DE LAS PROTEÍNAS 1KCS Y 1MBM

Medidas/Algoritmos	1KCS		1MBM	
	RT	RF	RT	RF
Precisión	0,41	0,48	0,3	0,28
Sensitividad	0,27	0,28	0,15	0,1

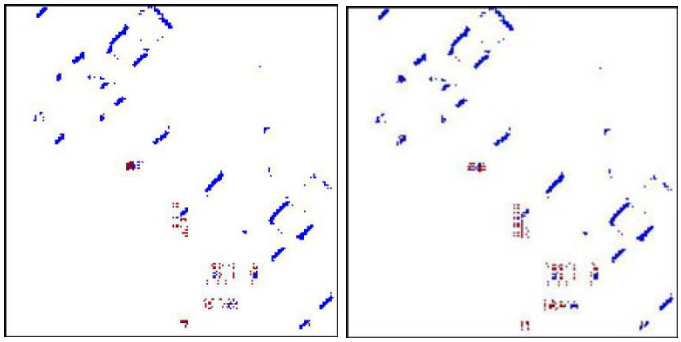
Como se puede observar los valores para la precisión en ambas proteínas se encuentran entre 0,28 y 0,48. Particularmente los valores son superiores en la proteína 1KCS, donde la precisión en supera el 0,40, En cuanto a la sensibilidad general, los valores se encuentran entre 0,10 y 0,28.



(a) **(b)**

Figura 5. Precisión y sensibilidad para las proteínas: (a) 1KCS y (b) 1MBM.

La Fig. 5a muestra un mejor comportamiento del algoritmo ***SPMCcm_FT*** sobre ***SPMCcm_RT***, para la proteína 1KCS, ***SPMCcm_RF***. En cambio, la Figura 5.b muestra todo lo contrario. Donde, para la proteína 1MBM, ***SPMCcm_RT*** reporta mejor comportamiento que ***SPMCcm_RF***. Por otro lado, ambos métodos presentan una sensibilidad similar para la proteína 1KCS, no resultando así para la proteína 1MBM. Como se puede apreciar los resultados son diversos con respecto a las dos proteínas, pero similares atendiendo a los algoritmos. Por lo que se puede concluir que ***SPMCcm*** se desempeña de manera similar frente a proteínas de aplicación real, con independencia del clasificador base que se haya escogido.



(a) **(b)**

Fig. 6. Mapa de contacto de la proteína 1kcs predicho por SPMCcm_RF (a) y SPMCcm_RT (b). Diagonal Superior representa la parte real, e inferior parte predicha.

En la Fig. 6 se muestran los mapas de contacto predichos por los métodos ***SPMCcm_RF*** y ***SPMCcm_RT***. En estas imágenes se puede apreciar que ambos métodos tienen un desempeño similar. La mayor diferencia se encuentra en la cantidad de falsos positivos, donde el ***SPMCcm_RT*** supera ligeramente a ***SPMCcm_RF***.

V. Conclusiones

En este artículo se presentó ***SPMCcm***, un algoritmo basado en la predicción de secuencias frecuentes y un esquema de clasificadores. El cual emplea la información brindada por la secuencia de aminoácidos, en dos etapas. Dónde, la primera etapa aprende de las interacciones entre las estructuras secundarias de las proteínas, las cuales extrae como secuencias frecuentes o ***itemsets***. Mientras, que la segunda etapa aprende de la interacción entre los aminoácidos presentes en las estructuras interactuantes o ***ítems***. Se realizó un proceso de evaluación experimental del algoritmo, tanto para diferentes dominios de aplicación (diferentes distribuciones de ***itemsets***), como en la validación externa en proteínas de Hepatitis C, empleando los clasificadores base de mejor y peor desempeño, con la intención de demostrar la efectividad del modelo propuesto. Como resultado, ***SPMCcm*** demostró que el algoritmo se comporta de forma similar, con independencia del clasificador base, alcanzando precisiones de hasta un 48%, superiores al 35% reportado por la literatura. Adicionalmente, al emplear árboles como clasificadores base, ***SPMCcm*** posee capacidad explicativa, útil para brindar pistas sobre el proceso de plegamiento de las proteínas.

AGRADECIMIENTOS

Esta investigación es una colaboración entre el Centro de Bioplantas y la Universidad de Ciego de Ávila, Cuba, la Universidad Católica de Santa María, Arequipa, Perú, y al Instituto Tecnológico de Ciudad Cuauhtémoc, Chihuahua, México. Especial agradecimiento a todo aquel que contribuyó con sus invaluables comentarios y consideraciones.

REFERENCIAS

[1] Manuscript Templates for Conference Proceedings, IEEE. http://www.ieee.org/conferences_events/conferences/publishing/templates.html

[2] Rakesh Agrawal and Ramakrishnan Srikant. Fast algorithms for mining association rules in large databases. In Jorge B. Bocca, Matthias Jarke, and Carlo Zaniolo, editors, Proceedings of the 20th International Conference on Very Large Data Bases, VLDB, pages 487-499, Santiago, Chile, September 1994.

[3] R. Agrawal, R. Srikant. Mining sequential patterns. IEEE Computer Society: In Proc. of the 11th International Conference on Data Engineering (ICDE' 95). Taipei, Taiwan, March, 1995.

[4] R Srikant, Agrawal R. Mining sequential patterns: Generalizations and performance improvements. In: 5th Intl. Conf. Extending Database Technology, 1996.

[5] J. Han, J. Pei, B. Mortazavi-Asl, Q. Chen, U. Dayal, M.-C. Hsu. FreeSpan: Frequent patternprojected sequential pattern mining. In Proc. 2000 Int.Conf. Knowledge Discovery and Data Mining, KDD'00. Boston, MA, Aug. 2000.

[6] Zaki MJ, Lesh N, Ogihara M (2000) PLANMINE: Sequence mining for plan failures. Artificial Intelligence Review, special issue on the Application of Data Mining.

[7] J. Xie, W. Ding, L. Chen, Q. Guo, y W. Zhang, «Advances in Protein Contact Map Prediction Based on Machine Learning», *Med. Chem.*, vol. 11, n.º 3, pp. 265–270, 2015.

[8] S. Mitra y Y. Hayashi, «Bioinformatics with soft computing», *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.*, vol. 36, n.º 5, pp. 616-635, sep. 2006.

[9] P. Fariselli y R. Casadio, «A neural network based predictor of residue contacts in proteins», Protein Eng., vol. 12, n.o 1, pp. 15–21, 1999.

[10] A. N. Tegge, Z. Wang, J. Eickholt, y J. Cheng, «NNcon: improved protein contact map prediction using 2D-recursive neural networks», Nucleic Acids Res., vol. 37, n.o Web Server, pp. W515-W518, jul. 2009.

[11] J. Cheng y P. Baldi, «Improved residue contact prediction using support vector machines and a large feature set», BMC Bioinformatics, vol. 8, n.o 1, p. 113, 2007.

[12] C. W. Howe y M. S. Mohamad, «Protein Residue Contact Prediction using Support Vector Machine», World Acad. Sci. Eng. Technol., vol. 60, pp. 1985–1990, 2011.

[13] H. Ashkenazy, R. Unger, y Y. Kliger, «Hidden conformations in protein structures», Bioinformatics, vol. 27, n.o 14, pp. 1941–1947, 2011.

[14] C. E. Santiesteban-Toca, G. Asencio-Cortés, A. E. Márquez-Chamorro, y J. S. Aguilar-Ruiz, «Short-Range interactions and decision tree-based protein contact map predictor», en Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics, Springer, 2012, pp. 224–233.

[15] P. Di Lena, K. Nagata, y P. Baldi, «Deep architectures for protein contact map prediction», Bioinformatics, vol. 28, n.o 19, pp. 2449-2457, oct. 2012.

[16] P. Chen y J. Li, «Prediction of protein long-range contacts using an ensemble of genetic algorithm classifiers with sequence profile centers», BMC Struct. Biol., vol. 10, n.o 1, p. 1, 2010.

[17] A. E. M. Chamorro, F. Divina, J. S. Aguilar-Ruiz, y G. A. Cortés, «A multi-objective genetic algorithm for the Protein Structure Prediction», en Intelligent Systems Design and Applications (ISDA), 2011 11th International Conference on, 2011, pp. 1086–1090.

[18] H. Drucker, C. Cortes, L. D. Jackel, Y. LeCun, and V. Vapnik, "Boosting and other ensembles methods," Neural Comput., vol. 6, pp. 1289–1301, 1994.

[19] W. Yan and K. Goebel, "Designing Classifier Ensembles with Constrained Performance Requirements," in Proceedings of SPIE Defense & Security Symposium, Multisensor Multisource Information Fusion: Architectures, Algorithms, and Applications, 2004.

[20] R. Maclin and J. Shavlik, "Combining the predictions of multiple classifiers: Using competitive learning to initialize neural networks," Fourteenth Int. Jt. Conf. Artif. Intell., pp. 524–531, 1995.

[21] P. Smyth, "Bounds on the mean classification error rate of multiple expert," Pattern Recognit. Lett., 1995.

[22] G. Giacinto and F. Roli, "Ensembles of Neural Networks for Soft Classification of Remote Sensing Images," Eur. Symp. Intell. Tech., pp. 166–170, 1997.

[23] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, "On combining Classifiers," IEEE Trans. Pattern Anal. Mach. Intell., vol. 2, no. 3, pp. 226–239, 1998.

[24] A. Tsymbal, M. Pecherizkiy, S. Puuronen, and D. W. Patterson, "Dynamic Integration of Classifiers in the Space of Principal Components," ADBIS, pp. 278–292, 2003.

[25] P. . Gislason, J. A. Benediktsson, and J. . Sveinsson, "Random Forest classification of multisource remote sensing and geographic data," IEEE Int. Geosci. Remote Sens. Symp. IGARSS'04, pp. 1049–1052, 2004.

[26] R. Jacobs, "Methods for combining experts' probability assessments," Neural Comput., vol. 7, pp. 867–888, 1996.

[27] G. I. Webb, "Multiboosting: a technique for combining Boosting and Wagging. Machine Learning," Kluwer Acad. Publ., vol. 40, pp. 159–196, 2000.

[28] R. S. Lynch and P. K. Willet, "Classifier fusion results using various open literature data sets," in EEE International Conference on Systems, Man and Cybernetics, pp. 723–728, 2003.

[29] K. A. Toh and W. Y. Yau, "Combination of hyperbolic functions for multimodal biometrics data fusion," IEEE Trans. Syst. Man Cybern., vol. 34, no. 2, pp. 1196–1209, 2004.

[30] L. Breiman, "Bagging predictors," Mach. Learn., vol. 24, no. 2, pp. 123–140, 1996.

[31] L. Breiman, "Random forests," Mach. Learn., vol. 45, no. 1, pp. 5–3, 2001.

[32] I. H. Witten and E. Frank, Data Mining: Practical Machie Learning Tools and Techniques, 2th ed. Morgan Kaufmann, p. 558, 2005.

[33] T. K. Ho, "The random subspace method for constructing decision forests," Tin Kam Ho. random Subsp. method IEEE Trans. Pattern Anal. Mach. Intell., vol. 20, no. 8, pp. 832–844, 1998.

[34] J. H. Friedman, "Stochastic gradient boosting. Computational Statistics and Data Analysis," Comput. Stat. Data Anal., vol. 38, pp. 367–378, 1999.

[35] Z. Zhou and Y. Yu, "Chapter 7. AdaBoost," in The Top Ten Algorithms in Data Mining, Taylor & Francis Group, LLC, pp. 127–149, 2009.

[36] G. I.Webb, "Multiboosting: A technique for combining boosting and wagging," Mach. Learn., vol. 40, no. 2, pp. 980–991, 2000.

[37] J. Gama and P. Brazdil, "Cascade generalization," Mach. Learn., vol. 41, no. 3, pp. 315–343, 2000.

[38] D. Wolpert, "Stacked generalization," Neural networks, vol. 5, pp. 241–260, 1992.

[39] A. K. Seewald and J. Fürnkranz, "An evaluation of grading classifiers," in 4th International Conference, IDA 2001, pp. 115–124, 2001.

[40] C. E. Brodley, "Recursive automatic bias selection for classifier construction," Mach. Learn., vol. 20, no. 1–2, pp. 63–94, 1995.

[41] S. J. Nowlan and G. E. Hinton, "Evaluation of adaptive mixtu res of competing experts," in Advances in Neural Information Processing Systems, pp. 774–780, 1990.

[42] P. Chan and S. J. Stolfo, “A comparative evaluation of voting and meta-learning on partitioned data,” in wel- fth International Conference on Machine Learning, pp. 90–98, 1995.
[43] R. Kohavi, “Scaling up the accuracy of naive-bayes classifiers: A decision- tree hybrid,” KDD, pp. 202–207, 1996.
[44] G. Pollastri y P. Baldi, «Prediction of contact maps by GIOHMMs and recurrent neural networks using lateral propagation from all four cardinal corners», Bioinformatics, vol. 18, n.o suppl 1, pp. S62–S70, 2002.
[45] A. Vullo y P. Frasconi, «A bi-recursive neural network architecture for the prediction of protein coarse contact maps», en Bioinformatics Conference, 2002. Proceedings. IEEE Computer Society, 2002, pp. 187–196.
[46] D. Baú, A. J. Martin, C. Mooney, A. Vullo, I. Walsh, y G. Pollastri, «Distill: a suite of web servers for the prediction of one-, two-and three-dimensional structural features of proteins», BMC Bioinformatics, vol. 7, n.o 1, p. 402, 2006.
[47] P. Di Lena, L. Margara, M. Vassura, P. Fariselli, y R. Casadio, «A new protein representation based on fragment contacts: towards an improvement of contact maps predictions», en Computational Intelligence Methods for Bioinformatics and Biostatistics, Springer, 2008, pp. 210–221.
[48] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, y F. Herrera, «A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches», *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.*, vol. 42, n.º 4, pp. 463-484, jul. 2012.
[49] V. Lopez, A. Fernandez, S. Garcia, V. Palade, y F. Herrera, «An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics», *Inf. Sci.*, vol. 250, pp. 113-141, nov. 2013.
[50] C. E Santiesteban-Toca, G. M Casanola-Martin, y J. S Aguilar-Ruiz, «A Divide-and-Conquer Strategy for the Prediction of Protein Contact Map», *Lett. Drug Des. Discov.*, vol. 12, n.º 2, pp. 124-130, 2015.
[51] C. E. S. Toca, M. García-Borroto, y J. S. A. Ruiz, «Using short-range interactions and simulated genetic strategy to improve the protein contact map prediction», en *Pattern Recognition*, Springer, 2012, pp. 166–175.
[52] N. Hamilton y T. Huber, «An introduction to protein contact prediction», *Bioinforma. Struct. Funct. Appl.*, pp. 87-104, 2008.
[53] P. Fariselli, O. Olmea, A. Valencia, y R. Casadio, «Prediction of contact maps with neural networks and correlated mutations», *Protein Eng.*, vol. 14, n.º 11, pp. 835–843, 2001.
[54] C. E. Santiesteban-Toca y J. S. Aguilar-Ruiz, «A new multiple classifier system for the prediction of protein’s contacts map», *Inf. Process. Lett.*, vol. 115, n.º 12, pp. 983-990, 2015.
[55] G. Zhang and K. Han, “Hepatitis C virus contact map prediction based on binary encoding strategy,” *Comput. Boiology Chem.*, vol. 31, pp. 233–238, 2007.
[56] E. Frank, M. Hall, L. Trigg, G. Holmes, and I. H. Witten, “Data mining in bioinformatics using Weka,” *Bioinformatics*, vol. 20, no. 15, pp. 2479–2481, 2004.